US-PAT-NO: 5870545

DOCUMENT-IDENTIFIER: US 5870545 A

TITLE: System and method for performing flexible workflow process compensation in a distributed workflow management system

---------- KWIC ---------

Abstract Text - ABTX (1):
A system and method for performing flexible workflow process compensation in a distributed workflow management system is described. A computer network includes a plurality of interconnected computers. Each computer includes a processor, memory and input/output facilities. The distributed workflow management system operates over the computer network. A plurality of **resources** perform the workflow process is performed with each **resource** operatively coupled to at least one of the computers. A process definition diagram includes computer-readable instructions stored in the memory of at least one of the computers and contains a role specification of process activities for performing the workflow process. A **resource** manager maps the role specification of process activities to at least one of the **resources** at runtime. A workflow process engine executes each process activity using the mapped **resource** and compensates a failed process activity responsive to the process definition diagram.

Brief Summary Text - BSTX (5):
This patent application is related to a commonly-assigned patent application entitled DISTRIBUTED WORKFLOW **RESOURCE** MANAGEMENT SYSTEM AND METHOD, Ser. No.
08/768261, now U.S. Pat. No. 5,826,239, filed on Dec. 17, 1996, the disclosure of which is incorporated herein by reference.

Brief Summary Text - BSTX (9):
While workflow process re-engineering provides a business management concept, workflow process management (WFPM) software-or more accurately, middleware-provides the enabling technologies for actually performing workflow process re-engineering. WFPM supports flexible solutions for the management of enterprise-wide operations, including workflow process control, automation and **monitoring; resource** allocation, authorization and authentication; task initialization and data exchange; and end-to-end communication and security. However, while WFPM offers an overall environment and approach to unifying, automating and **measuring** workflow processes, it is not limited to supporting workflow process re-engineering and can be used to manage existing nonautomated legacy or work processes.

Brief Summary Text - BSTX (10):

In general, WFPM systems perform a wide range of tasks. For instance, they can provide a method for defining and managing the flow of a work process or support the definition of **resources** and their attributes. In addition, they can assign **resources** to work, determine which steps will be executed next within a work process and when they will be executed and can ensure that the workflow process continues until proper termination. Moreover, they can notify **resources** about pending work, enforce administrative policies, such as access control and track execution and support user inquiries of status. Finally, they can provide history information in the form of an audit trail for completed workflow processes and collect statistical data for process and **resource** bottleneck analysis, flow optimization and automatic workload balancing.

Brief Summary Text - BSTX (12):

In a WFPM system, some applications require the selective use of transactional properties for individual tasks or entire workflow processes to meet consistency or reliability requirements of the business. Compensation is a useful way to ensure these transactional properties for workflow processes. Although compensation is also used in database applications, the compensation of workflow processes is very different from compensation of database transactions for the following reasons. First, a workflow process is structurally more complex than a database transaction. For instance, a workflow process usually accesses multiple independent **databases and other resources** and systems. Additionally, it is difficult to implement a workflow process as a single transactional unit or as a simple sequence of transactional units. As a result, specifying the compensation scope for a workflow process becomes a non-trivial task.

Brief Summary Text - BSTX (15):

There is a need for a system and method for flexibly and efficiently compensating a structurally complex workflow process. Such a workflow process could consists of an arbitrarily complex sequence of tasks and involve a plurality of **databases, resources** and systems.

Brief Summary Text - BSTX (20):

An embodiment of the present invention is a system and method for performing flexible workflow process compensation in a distributed workflow management system. A computer network includes a plurality of interconnected computers. Each computer includes a processor, memory and input/output facilities. The distributed workflow management system operates over the computer network. A plurality of **resources** perform the workflow process is performed with each **resource** operatively coupled to at least one of the computers. A process definition diagram includes computer-readable instructions stored in the memory of at least one of the computers and contains a role specification of process activities for performing the workflow process. A **resource** manager maps the role specification of process activities to at least one of the **resources** at runtime. A workflow process engine executes each process activity using the mapped **resource** and compensates a failed process activity responsive to the process definition diagram.

Detailed Description Text - DETX (3):

FIG. 1 shows a block diagram of a workflow process management (WFPM) system 10 implemented in a network 11 of computer systems 12a-d coupled to a plurality of users 14a-b and machines 15a-b for management and control of workflow process activities. Each computer system 12a-d is shown coupled with a single user 14a-b or machine 15a-b, but multiple users or machines or combinations thereof can also be employed. The WFPM system 10 is shown from an enterprise perspective with the control and coordination of each of the computer systems 12a-d being accomplished by computer software, preferably object-oriented software, executed as a distributed application by the computer systems 12a-d. Optionally, workflow process activity information, such as **resource** data and rules, can be stored in a database on a centralized WFPM server 17 which is accessible by the computer systems 12a-d over the network 11 or can be stored in a plurality of databases on each of the computer systems 12a-d. The computer systems 12a-d and centralized WFPM server 17 conventionally include a processor, memory and input/output interface including network communications facilities and user input and output devices.

Detailed Description Text - DETX (4):

Each workflow process 18 includes a sequence of activities, each of which is ordinarily performed by one of the computer systems 12a-d in conjunction with an associated user 14a-b or machine 15a-b, although some activities can be performed by microprocessor-controlled devices 16 (one such device shown in FIG. 1, although multiple devices can be used), such as a telephone or facsimile machine, printing device or similar self-controlling mechanism. In addition, each machine 15a-b can be a work instrument or computer **resource**.

Detailed Description Text - DETX (5):

The workflow process 18 can span several business organizations (only one organization is shown in FIG. 1) with multiple activities potentially performed in parallel. In such cases, the WFPM system 10 acts as the "superstructure" that ties together disparate computer systems 12a-d whose business purposes are interconnected. The WFPM system 10 provides procedural automation 13 of the workflow process 18 by managing the sequence of process activities and the invocation of appropriate user 14a-b, machine 15a-b or microprocessor-controlled device 16 **resources** associated with the various activity steps.

Detailed Description Text - DETX (7):

The procedural automation 13 of the workflow process 18 involves the high-level specification of individual workflows (examples shown in FIG. 3 and FIG. 7) which provides the operational "glue" and environment support needed by the WFPM system 10 for managing and automating the workflow processes 18, recovering from failures and enforcing consistency. As further described hereinbelow, the WFPM system 10 also enforces various administrative policies associated with **resources** and work.

Detailed Description Text - DETX (11):
Briefly, HP OpenPM provides an open system adhering to the CORBA communications infrastructure with a Workflow Management Coalition-standard interface. Second, it offers high performance as a result of **optimized database** access and commitment features. It also provides effective management when coupled with an HP OpenView-based system management environment. Finally, HP OpenPM presents a comprehensive solution for business re-engineering, including an extensive set of products.

Detailed Description Text - DETX (12):
The overall architecture of the HP OpenPM system is depicted in FIG. 2. The core is the HP OpenPM engine 20, which supports five interfaces. The interfaces enable the HP OpenPM engine 20 to interact with workflow process designer 22a-c, workflow process instance execution 23a-b, workflow process **monitor** 24a-c, workflow management 28a-c and business object management modules 30, 31, 32, 33. In addition, worldwide web client support is provided by each individual network node 12a which can execute middleware modules expressed in platform-independent languages, such as Java Applets and HTML code. An HP OpenPM database 21 is maintained on the centralized WFPM server 17 (shown in FIG. 1) for use by the HP OpenPM engine 20.

Detailed Description Text - DETX (13):
A workflow process 18 is specified by the process design modules 22a-c via the workflow process definition interface. An instance of a workflow process 18 can be started, controlled or stopped by the process instance execution modules 23a-b via the process execution interface. Status information of each process instance and load information for the WFPM system 10 can be queried using the process status **monitor** modules 24a-c via the process status monitoring interface. The workflow management interface is used to allocate, at run time, execution **resources** to a task, according to the policies defined by the organization (including authorization and authentication) and the availability of the **resources** using the workflow management modules 28a-c. Interaction with the external world, such as invoking an application, controlling an instrument or delivering a work order to a person's electronic mail in-box, is performed by the various business object management modules 30, 31, 32, 33.

Detailed Description Text - DETX (19):
Work nodes 41, 43, 45, 46, 48, 50, 52, 54 represent activities to be performed external to the HP OpenPM engine 20. These activities include authorization, **resource** allocation, execution of business objects 93a-c and provision of input data for the business objects 93a-c and output data from them. Rule nodes 42, 44, 47, 49, 51, 53, 55 represent processing internal to the HP OpenPM engine 20. This processing includes decisions of about which nodes should execute next, generation or reception of events, and simple data manipulation.

Detailed Description Text - DETX (31):
To **monitor** the progress of running process activities and support system

management, the HP OpenPM engine 20 maintains a comprehensive log of all events using a log manager 70 and provides a native interface 79a as well as an SNMP 79b and CMIP 79c gateways to facilitate integration with the HP OpenView environment. The formats and contents of the logged information can be customized to support specific application needs.

Detailed Description Text - DETX (35):
Based on CORBA technology, in the HP OpenPM engine 20, an abstraction called a business object 93a (shown in FIG. 5) is built to encapsulate whatever piece of work each process activity has to accomplish. The wrapping code provides an IDL (Interface Definition Language) interface. The business objects are catalogued by a database manager 64 in the HP OpenPM business object library in business databases 94a-c (shown in FIG. 5). An object cache 75 is optionally used to **optimize** business object access.

Detailed Description Text - DETX (38):
HP OpenPM **Resource** and Policy Management

Detailed Description Text - DETX (39):
A **resource** is a person, computer process or machine that can be used to accomplish a task. A **resource** has a name and various attributes defining its characteristics, such as job code, skill set, organization unit and availability.

Detailed Description Text - DETX (40):
A policy is a set of rules that determines how **resources** are related to tasks within a WFPM system 10. One common use is for task assignment. Policies can be used to specify which **resource,** under which role, is eligible or available to perform a task. Policies are also used to ensure proper authorization and authentication.

Detailed Description Text - DETX (41):
In HP OpenPM, the mapping between the process activity (task) specified in a workflow process 18 and the business object (**resource**) to be invoked is performed by the **resource** manager 28a (shown in FIG. 2) during run time as part of the execution of the process activity. The HP OpenpM engine 20 allows multiple **resource** managers 28a-c to be used to resolve a single **resource** assignment request; each resolves the request at a different level within an organization.

Detailed Description Text - DETX (51):
At the service management layer 102, the WFPM process enabling framework is required to be able to support re-engineering and transformation processes for strategic operations support systems and business support systems, to integrate existing operational environments to form an enterprise hub for service management and provisioning, deploy new management services as rapidly as possible, to **monitor and measure** processes, to tune processes to benefit from

experience and to automate processes to reduce execution time.

Detailed Description Text - DETX (54):
The scenario demonstrated by this prototype consists of the provision of a new VI4/VC12 path for customers. It goes through several different steps for this operation: search for a new route, negotiate the **service level agreement (SLA)** with the customer, configure the new path, and finally, update the **SLA** for this customer. The HP OpenPM process definition supporting the process of providing this new SONET data path is sketched in FIG. 7 which shows the HP OpenView process definition for SONET configuration management.

Detailed Description Text - DETX (61):
4. If not, issue a request for an engineer to go on-site and physically connect the endpoints to the end-ADMs. After the establishment of the connection, the process continues on to step 5 and an independent subprocess is initiated to watch for **resource** changes.

Detailed Description Text - DETX (62):
5. Find valid routes between end-ADMs. This requires access to the routing table in the **SLA database** to determine whether any valid routes exist between the two end-ADMs. Either a list of ADMs is returned signifying the ADMs that must be configured to realize the route, or "No Route Found" is returned. For a returned list of ADMs, this activity will then use the HP OpenView DM facility agent to collect port information stored in the MIB to determine the available ports between the ADMs that are fibered together and can be used to enable the path.

Detailed Description Text - DETX (63):
6. Check network element (NE) capabilities. For an ADM in the route, this activity uses the HP OpenView DM NE agent to access the MIB information to determine whether a VC4 cross-connection can be set up in the ADM between the selected ports of the ADM. This activity has to be executed for each ADM in the route. During steps 5 and 6, if any additional **resources** become available, HP OpenPM cancels any currently running activity and starts the process over from step 5 to consider these newly available **resources**.

Detailed Description Text - DETX (65):
8. Configure the selected route. This activity is responsible for setting up the cross-connections in each ADM by invoking the HP OpenView DM NE agent and updating the **SLA database**.

Detailed Description Text - DETX (67):
FIG. 8 is, by way of example, a process definition diagram for a synchronous digital hierarchy (SDH) network configuration management process 149 for use with the system architecture of FIG. 5, such as described in J. Davis et al., "Flexible Compensation of OpenPM Workflow Processes," HP Labs. Tech. **Report,** HPL-96-72, May 1996, the disclosure of which is incorporated herein by

reference for all purposes. Compensation of workflow processes involve undoing already completed transactions if not all transactions complete successfully. It is a complicated task that involves process designers' effort in part because business objects are too complex to be fully compensated automatically and different processes have different compensation needs which only process designers understand.

Detailed Description Text - DETX (94):
Referring back to FIG. 5, a process 149 is defined using a process designer 22a. Once defined, the process 149 can be started, stopped or intervened using a process controller 90a. A process **monitor** 24a keeps status information about each individual process 149 and loads information about the entire system. Each **resource** manager 28a has access to the information about **resources** usable by a process 149. At runtime, the **resource** manager 28a maps **resources** to process activities to perform tasks. The mapping is performed according to the role specification of process activities and the policies provided by the **resource** administrator. In addition, the **resource** manager 28a can consult other data sources in fulfilling requests, for example, a roster indicating who is in which department or on vacation.

Claims Text - CLTX (3):
mapping the role specification of process activities using a **resource** manager to at least one of the **resources** at runtime;

Claims Text - CLTX (4):
executing each process activity with the mapped **resource** using a workflow process engine, including forming a pre image storing a portion of process relevant data provided to the process activity and a post image storing a portion of process relevant data generated by the process activity;

Claims Text - CLTX (6):
re-executing at least one further such process activity with the mapped **resource** using a workflow process engine if the at least one further such process activity is in the lazy compensation group responsive to the process definition diagram.

Claims Text - CLTX (19):
12. A method using a workflow process engine for flexibly compensating a workflow process in a distributed workflow management system, the distributed workflow management system comprising a computer network including a plurality of interconnected computers operatively coupled to a plurality of **resources,** each computer including a processor, memory and input/output facilities, the method comprising the steps of:

Claims Text - CLTX (22):
performing a forward activity for performing each process activity in the workflow process using one of the **resources**;

Claims Text - CLTX (56):
mapping the role specification of process activities using a **resource** manager to at least one of the **resources** at runtime;

Claims Text - CLTX (58):
executing each process activity with the mapped **resource** using a workflow process engine, including forming an original pre image of each such process activity before execution and an original post image of each such process activity after execution; and

Claims Text - CLTX (73):
mapping the role specification of process activities using a **resource** manager to at least one of the **resources** at runtime; and

Claims Text - CLTX (74):
executing each process activity with the mapped **resource** using a workflow process engine; and

Claims Text - CLTX (91):
a plurality of **resources** upon which the workflow process is performed, each **resource** being operatively coupled to at least one of the computers;

Claims Text - CLTX (93):
a **resource** manager mapping the role specification of process activities to at least one of the **resources** at runtime; and

Claims Text - CLTX (94):
a workflow process engine executing each process activity using the mapped **resource** and compensating a failed process activity responsive to the process definition diagram.

Claims Text - CLTX (95):
32. A system according to claim 31, wherein the process definition diagram comprises a set of work nodes and rule nodes interconnected by unidirectional arcs, each work node being associated with one of the process activities, each rule node being associated with at least one route through the process definition diagram, the workflow process engine evaluating each work node to perform the associated process activity using one of the **resources** and each rule node to follow the associated at least one route through the process definition diagram.